IoT Smart Water Management System

👥 *Yahya Al Benni – Oussama Khatib – Elias Jamhour*

Introduction

- IoT devices need efficient communication protocols for data exchange.
- Two commonly used protocols for constrained devices: **MQTT** (Message Queuing Telemetry Transport) and **CoAP** (Constrained Application Protocol).
- This presentation compares both and explains their **role in monitoring the main tank water level, house tank water level, turbidity sensor, user notifications, and pump control** in our project.

MQTT Overview

What is MQTT?

- A lightweight, **publish-subscribe** messaging protocol.
- Uses a **broker** to relay messages between devices.
- Designed for low-bandwidth, high-latency networks.

How it Works:

1. A device **publishes** data to a topic.
2. A broker **receives** the message and distributes it.
3. Other devices **subscribe** to the topic to receive updates.

Advantages:
✅ Low bandwidth usage.
✅ Reliable message delivery (QoS levels).
✅ Works over TCP/IP, ensuring stable communication.

CoAP Overview

What is CoAP?

- A **request-response** protocol similar to HTTP but optimized for IoT.
- Uses **UDP** instead of TCP for lightweight operation.
- Designed for constrained networks and low-power devices.

How it Works:

1. A client **sends** a request to a server (e.g., "GET water level").
2. The server **responds** with the requested data.
3. Supports **confirmable** and **non-confirmable** messages.

Advantages:
✅ Works efficiently with resource-constrained devices.
✅ Faster response time due to UDP.
✅ Supports multicast communication.

Comparison (MQTT vs. CoAP)

Feature	MQTT	CoAP
Communication	Publish-Subscribe	Request-Response
Transport Protocol	TCP	UDP
Reliability	High (QoS levels)	Lower (optional acknowledgments)
Bandwidth Usage	Low	Very Low
Latency	Moderate	Low
Best For	Real-time monitoring	Command-response IoT interactions

---

Implementation in Our Project

Both **MQTT and CoAP** were used to monitor the **main tank water level, house tank water level, turbidity sensor**, provide **user notifications**, and enable **pump control**.

MQTT:

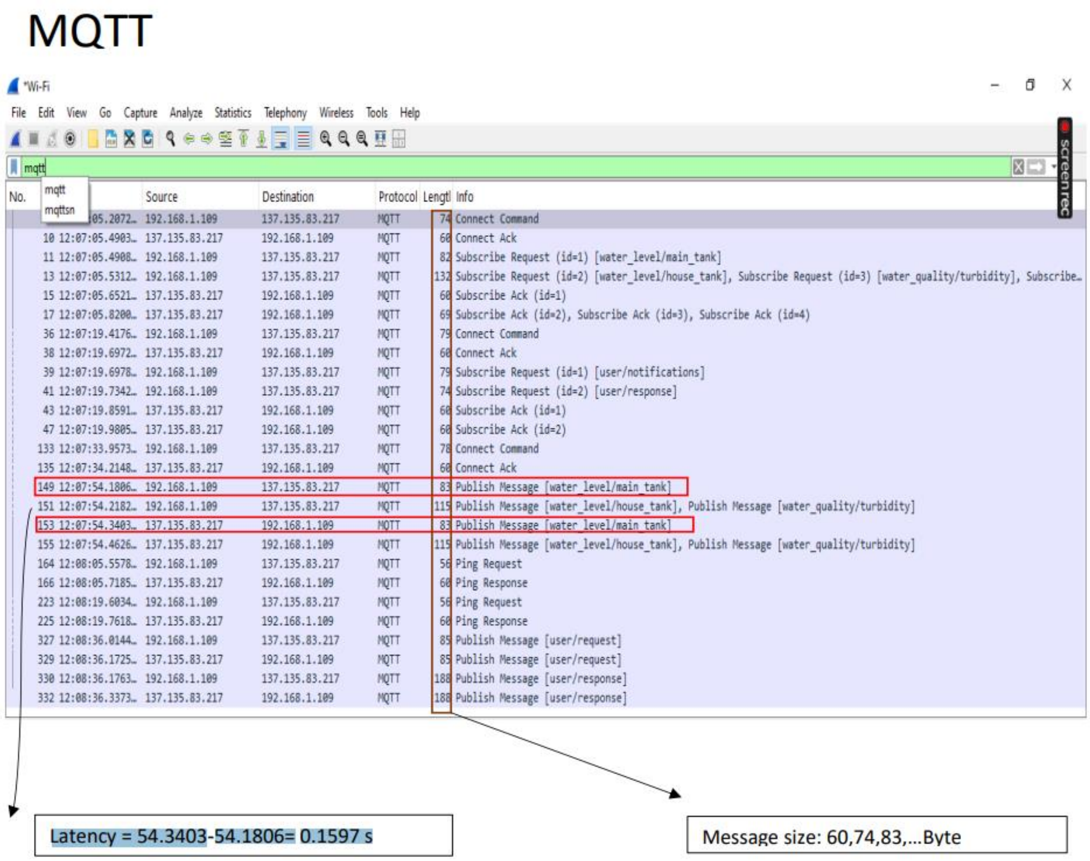
- The **main tank and house tank sensors** publish real-time water levels.
- The **turbidity sensor** publishes water quality data.
- The **mobile app subscribes** to receive **real-time notifications** when:
  - **House tank level is low** (user is alerted to reduce water usage).
  - **Turbidity is high** (user is warned about water quality).
  - **Main tank is full** (pump stops automatically).

CoAP:

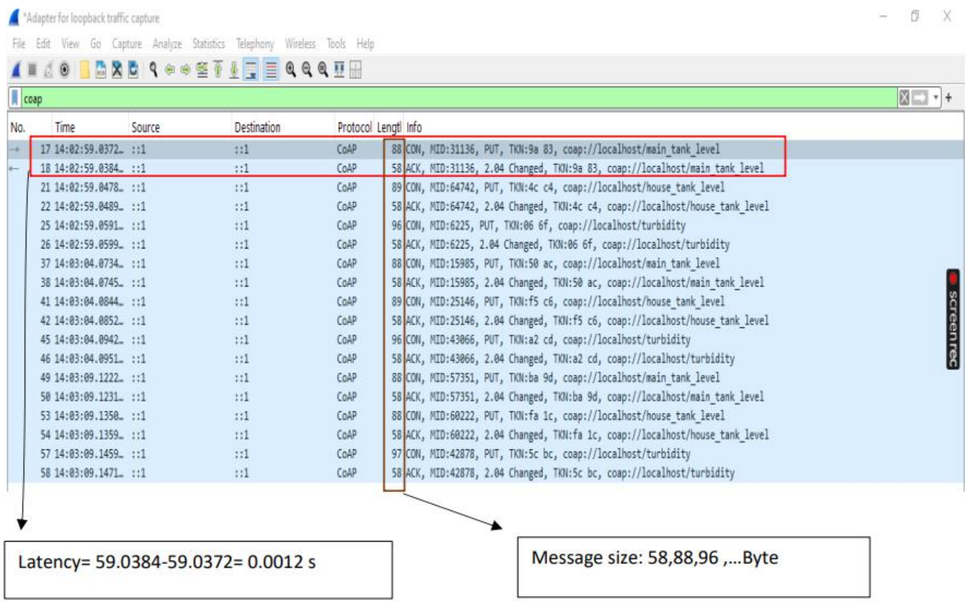
- The **pump controller requests** sensor data when needed.
- The **server responds** with:
  - **Main tank water level**
  - **House tank water level**

- o **Turbidity status**
- The system **triggers pump control actions** based on sensor feedback.

Results and Observations

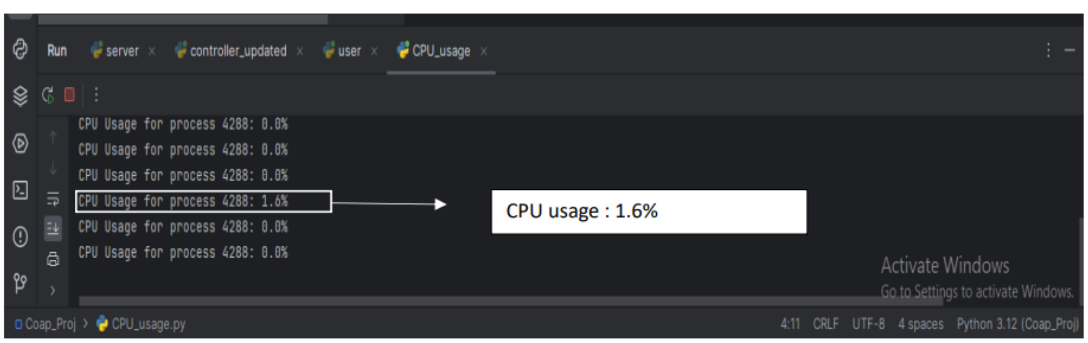


Coap



MQTT CPU usage (between 2.0% and 9.7%)

Coap CPU usage



- **CoAP is more efficient** for sensor data requests due to **lower CPU usage and latency**.
- **MQTT is better for continuous monitoring** since it can provide real-time updates without needing repeated requests.

- **Message sizes in MQTT are larger** due to additional headers, but this is acceptable for non-constrained networks.